IN THE CLAIMS

Please amend claim 15 and cancel claims 1-4, 6-7, 11-14 and 16 (without prejudice to including them in a later-filed continuation application), as follows:

Listing of Claims:

1-7.   (Cancelled)

8.   (Previously Amended) A computer-implemented method, comprising:

creating one or more data structures that together store characteristics of a plurality of active branches B^{active} that make up a graph of nodes and branches that form a circuit, wherein B^{active} consists of

a set B^L of zero or more inductive branches, each having a non-zero inductive component but neither a capacitive component nor a variable switch state;

a set B^C of zero or more capacitive branches, each having a non-zero capacitive component but neither an inductive component nor a variable switch state; and

a set B^A of additional branches, each having neither an inductive component, nor a capacitive component;

partitioning B^{active} into a first branch set B^{active}_{tree} and a second branch set B^{active}_{link}, where the branches in B^{active}_{tree} form a spanning tree over B^{active}, giving priority in said partitioning to branches not in B^L over branches in B^L;

sub-partitioning B^{active}_{link} into a third branch set B^L_{link} and a fourth branch set B^{CA}_{link}, where

$$B^L_{link} = B^{active}_{link} \cap B^L;$$

identifying a fifth branch set B^{CA} as the union of

Amendment
Application No. 10/043,981

Page 2 of 7
Group Art Unit 2128
CIRCUIT SIMULATION

Inventors: Wasynczuk et al.
Attorney Docket No. 31122-12.

B_{link}^{CA} ,

$B^C \cap B_{tree}^{active}$, and

those branches in B_{tree}^{active} that form a closed graph when combined with B_{link}^{CA} ;

partitioning B^{CA} into a sixth branch set $\widetilde{B}_{tree}^{CA}$ and a seventh branch set $\widetilde{B}_{link}^{CA}$, where the

branches in $\widetilde{B}_{tree}^{CA}$ form a spanning tree over B^{CA} , giving priority in said partitioning to branches

in B^C over branches not in B^C ;

identifying an eighth branch set $B_{tree}^C = \widetilde{B}_{tree}^{CA} \cap B^C$;

selecting a set of state variables comprising:

for each branch of B_{link}^L , either the inductor current or inductor flux, and

for each branch of B_{tree}^C , either the capacitor voltage or capacitor charge; and

simulating a plurality of states of the circuit using the set of state variables.

9.   (Original) The method of claim 8, wherein said partitioning steps each comprise
an application of a weighted spanning tree algorithm.

10.   (Original) The method of claim 9 wherein, for some positive numbers w_L and w_C:
for the partitioning of B^{active} , a minimum spanning tree algorithm is used with weight

function $\omega_L(b_j) = \begin{cases} w_L \ \text{if branch } b_j \in B^L \\ 0 \ \text{otherwise} \end{cases}$; and

for the partitioning of B^{CA} , a maximum spanning tree algorithm is used with weight

function $\omega_C(b_j) = \begin{cases} w_C \ \text{if branch } b_j \in B^C \\ 0 \ \text{otherwise} \end{cases}$.

11-14. (Cancelled)

Amendment   Page 3 of 7   Inventors: Wasynczuk et al.
Application No. 10/043,981   Group Art Unit 2128   Attorney Docket No. 31122-12
                                   CIRCUIT SIMULATION

PAGE 6/10 * RCVD AT 10/24/2006 3:49:11 PM [Eastern Daylight Time] * SVR:USPTO-EFXRF-6/38 * DNIS:2738300 * CSID:317 637 7561 * DURATION (mm-ss):01-52

15.   (Presently Amended) <u>A system, comprising a processor and a computer-readable</u>
<u>medium in communication with said processor, said medium containing programming</u>
<u>instructions executable by said processor to:</u>

<u>build state equations for a first topology of an electronic circuit having at least two switching</u>

<u>elements, wherein each switching element has a switching state;</u>

<u>solve said state equations at time t_i to provide a state output vector, in which at least two</u>

<u>elements control the switching states of the switching elements;</u>

<u>calculate the value of a switching variable as a function of the state output vector, wherein</u>

<u>the value reflects whether the switching state of at least one of the switching elements is</u>

<u>changing; and</u>

<u>if the value of the switching variable at time t_i indicates that at least one of the switching</u>

<u>elements is changing, determine a second topology of the electronic circuit for time t_i^+ and obtain</u>

<u>state equations for the second topology;</u>

<u>wherein:</u>

<u>said programming instructions comprise a state equation building module, a solver</u>

<u>module for ordinary differential equations, and a switching logic module;</u>

<u>said building is performed by the state equation building module;</u>

<u>said solving and calculating are performed by the solver module;</u>

<u>said determining is performed by the switching logic module;</u>

~~The system of claim 12, wherein:~~
at a time t_j, at least two switching elements are each either rising-sensitive or falling-

sensitive switches, wherein

rising-sensitive switches change switching state if and only if a controlling element of

the state vector has passed from a negative value to a non-negative value; and

Amendment	Page 4 of 7	Inventors: Wasynczuk et al.
Application No. 10/043,981	Group Art Unit 2128	Attorney Docket No. 31122-12
	CIRCUIT SIMULATION	

PAGE 7/10 * RCVD AT 10/24/2006 3:49:11 PM [Eastern Daylight Time] * SVR:USPTO-EFXRF-6/38 * DNIS:2738300 * CSID:317 637 7561 * DURATION (mm-ss):01-52

falling-sensitive switches change switching state if and only if a controlling element of

the state vector has passed from a positive value to a non-positive value; and

the function is the arithmetic maximum of

a maximum of all controlling elements of the state vector that control rising-sensitive

switches, and

the negative of the minimum of all controlling elements of the state vector that control

falling-sensitive switches.

16.   (Cancelled)